Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku

**Milan Milinčević**

# Metode digitalnog potpisivanja

Diplomski rad

Osijek, 2019.

Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku

**Milan Milinčević**

# Metode digitalnog potpisivanja

Diplomski rad

Voditelj: izv. prof. dr. sc. Ivan Matić

Osijek, 2019.

# Sadržaj

# Uvod

Za potrebe komunikacije često se koriste poruke koje sa sobom nose potpise. Problem koji nastaje kod potpisivanja poruke jest je li taj potpis moguće krivotvoriti i je li on autentičan. Kao alati za bolju vjerodostojnost potpisa koriste se brojne sheme digitalnog potpisa među kojima su RSA shema potpisa, Rabinova shema, ElGamalova itd. Algoritmi, odnosno sheme potpisa jasno definiraju način potpisivanja i otkrivanja potpisa prilikom čega je potrebno poznavati javni i tajni ključ. Navedene se sheme potpisa oslanjaju na teško rješive probleme te tako osiguravaju sigurnost potpisa. Neki od tih problema su problem faktorizacije te problem diskretnog logaritma.

U prvom dijelu rada ćemo nešto reći o digitalnom potpisu, a zatim ćemo se baviti kriptosustavima zasnovanim na problemu faktorizacije te njihovim shemama potpisa. Reći ćemo nešto više o RSA te Rabinovoj shemi potpisa. Zatim ćemo se u drugom poglavlju zadržati na DSA shemi potpisa. Spomenut ćemo problem diskretnog logaritma te ElGamalov kriptosustav koji se upravo na tome zasniva. U natavku ćemo definirati ElGamalovu shemu potpisa te zatim DSA algoritam koji je nastao prema uzoru na ElGamalovu shemu potpisa, odnosno ElGamalov kriptosustav. Nakon definiranja potrebnih pojmova te su sheme ilustrirane na primjerima na kojima je vidljiv proces potpisivanja te provjere autentičnosti potpisa. U posljednjim dijelovima rada definirane su nepobitne te fail-stop sheme potpisa. Nepobitne sheme potpisa imaju neke nove značajke u odnosu na već pomenute sheme i one su posebno naglašene u radu.

# Poglavlje 1

# Digitalni potpis

Većina ljudi koristi potpis u raznim svakodnevnim situacijama kao što su podizanje paketa u pošti, potpisivanje raznih ugovora itd. Glavna svrha potpisa jest identificirati odgovornu osobu. Svakako bi bila želja prenijeti svojstva potpisa i u digitalni svijet. Upravo zbog toga nastaje digitalni potpis koji povezuje poruku s originalnim subjektom. Shema digitalnog potpisa je metoda potpisivanja poruke spremljene u elektronskoj formi. Shema potpisa sastoji se od algoritma za generiranje potpisa te algoritma za provjeru istog. Prije nego se pozabavimo shemama potpisa pokušat ćemo odgovoriti na par pitanja koja nam se nameću spominjanjem digitalnog potpisa.

Prvo pitanje je pitanje provjere. Klasični potpis se provjerava tako da se usporedi s autentičnim potpisom. Na primjer, nakon referenduma na kojemu se skupljaju potpisi građana treba provjeriti vjerodostojnost svih potpisa. To se radi tako da se potpisi uspoređuju s onima na službenim dokumentima. Međutim, jasno je da to nije sigurna metoda jer se vrlo lako može krivotvoriti nečiji potpis. S druge strane, digitalni potpis se vrlo lako može provjeriti pomoću javnog algoritma za provjeru istog. Kako je algoritam javan bilo tko može provjeriti digitalni potpis.

Drugo pitanje na koje želimo odgovoriti je svakako potpisivanje dokumenta. Kod klasičnog potpisivanja potpis je fizički dio dokumenta. No kod digitalnog potpisivanja situacija je drugačija. Digitalni potpis nije fizički vezan za poruku koju treba potpisati. Prema tome, algoritam koji koristimo mora znati nekako povezati poruku s pripadajućim potpisom.

Kod digitalnog potpisa se kopija potpisane poruke ne razlikuje od originala, dok kod klasičnog potpisa dva potpisana dokumenta se mogu razlikovati. Problem identičnosti kopije i originala kod digitalnog potpisa može biti sprječavanje iskorištavanja potpisa samo jednom. Na primjer, Marko želi pokloniti Ivi kupon za kupovinu preko interneta u određenom iznosu. Kupon joj je poslao preko maila koristeći digitalni potpis kako bi

ga samo ona mogla iskoristiti. Također, Marko želi da Iva može kupon iskoristiti samo jednom. Zbog toga, digitalna poruka, u našem primjeru kupon, treba sadržavati neke dodatne informacije za onemogućavanje ponovne upotrebe, na primjer, datum.

Kao što smo već spomenuli, shema potpisa se sastoji od dva dijela; algoritma za generiranje potpisa te algoritma za provjeru tog potpisa. Marko može potpisati poruku $x$ koristeći (privatni) algoritam za generiranje potpisa $sig_K$ koji ovisi o $K$, pri čemu je $K$ neki tajni ključ. Primjenom javnog algoritma za provjeru, $ver_K$, možemo provjeriti dobiveni potpis, $sig_K(x)$. Za uređeni par $(x, y)$, pri čemu je $x$ poruka, a $y$ navodni potpis poruke $x$, algoritam provjere kao rezultat vraća boolean vrijednost "istina" ili "laž" ovisno o tome je li $y$ važeći potpis za poruku $x$ ili nije.

**Definicija 1.1.** *Shema potpisa je uređena petorka $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{S}, \mathcal{V})$ pri čemu vrijedi:*

1. *$\mathcal{P}$ je konačan skup svih mogućih poruka*
2. *$\mathcal{C}$ je konačan skup svih mogućih potpisa*
3. *$\mathcal{K}$ je skup svih mogućih ključeva*
4. *Za svaki $K \in \mathcal{K}$ postoji algoritam potpisa $sig_K \in \mathcal{S}$ i pripadajući algoritam provjere $ver_K \in \mathcal{V}$.*
   *Funkcije $sig_K : \mathcal{P} \to \mathcal{C}$ i $ver_K : \mathcal{P} \times \mathcal{C} \to \{istina,\ laž\}$ za svaku poruku $x \in \mathcal{P}$ i za svaki potpis $y \in \mathcal{C}$ moraju zadovoljavati sljedeće:*

$$ver_K(x, y) = \begin{cases} istina, & y = sig_K(x) \\ laž, & y \neq sig_K(x) \end{cases}.$$

   *Potpisana poruka je uređeni par $(x, y)$ pri čemu je $x \in \mathcal{P}$ te $y \in \mathcal{C}$.*

Kao što smo već spomenuli $ver_K$ je javna funkcija, dok je $sig_K$ privatna. Odnosno, $sig_K$ je funkcija koja treba biti poznata samo osobi koja potpisuje poruku. Vratimo li se na prošli primjer u kojemu Marko Ivi šalje kupon, nitko osim Marka ne bi trebao moći izračunati potpis $y$ tako da vrijedi $ver_K(x, y) = \text{istina}$. Potpis $y$ nazivamo krivotvorenim ukoliko neka treća osoba uspije odrediti uređeni par $(x, y)$ pri čemu vrijedi $ver_K(x, y) = \text{istina}$ te poruku $x$ nije potpisao Marko. Odnosno, krivotvoreni potpis je validan potpis koji je potpisao netko drugi umjesto Marka. U nastavku ćemo se pozabaviti RSA kriptosustavom, tj. RSA shemom potpisa koja se može iskoristiti za dobivanje digitalnog potpisa.

# 1. RSA shema potpisa

Prije nego se pozabavimo RSA shemom potpisa potrebno je definirati pojmove kao što su kriptosustav te RSA kriptosustav.

**Definicija 1.2.** *Kriptosustav je uređena petorka* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *pri čemu vrijedi:*

1. $\mathcal{P}$ *je konačan skup svih mogućih osnovnih elemenata otvorenog teksta*
2. $\mathcal{C}$ *je konačan skup svih mogućih osnovnih elemenata šifrata*
3. $\mathcal{K}$ *je skup svih mogućih ključeva*
4. $\mathcal{E}$ *je skup svih mogućih funkcija šifriranja*
5. $\mathcal{D}$ *je skup svih mogućih funkcija dešifriranja*
6. *Za svaki* $K \in \mathcal{K}$ *postoji funkcija šifriranja* $e_K \in \mathcal{E}$ *i pripadajuća funkcija dešifriranja* $d_K \in \mathcal{D}$. *Funkcije* $e_K : \mathcal{P} \to \mathcal{C}$ *i* $d_K : \mathcal{C} \to \mathcal{P}$ *zadovoljavaju svojstvo:*

$$d_K(e_K(x)) = x, \quad \forall x \in \mathcal{P}.$$

Kriptosustavi se obično dijele prema sljedećim kriterijima: [1]

1. Tip operacija koje koristimo tijekom šifriranja (npr. transpozicijske šifre)
2. Način prema kojem obrađujemo otvoreni tekst (npr. blokovne šifre)
3. Javnost i tajnost ključeva (npr. simetrični kriptosustavi).

Jedan od kriptosustava s javnim ključem je upravo i RSA kriptosustav koji je zasnovan na problemu faktorizacije velikih prirodnih brojeva. Nastao je 1977. godine, a nazvan je po svojim tvorcima Ronaldu **R**ivestu, Adi **S**hamiru te Leonardu **A**dlemanu.

[1] više o ovome se može pronaći u [1]

**Definicija 1.3.** *Neka vrijedi $n = pq$ pri čemu su $p$ i $q$ prosti brojevi, te $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n = \{0, 1, \ldots, n-1\}$. Neka je*

$$\mathcal{K} = \{(n, p, q, d, e) : n = pq, \quad de \equiv 1 \pmod{\varphi(n)}\}.$$

*Za $K \in \mathcal{K}$ definiramo:*

$$e_K(x) = x^e \mod n,$$
$$d_K(y) = y^d \mod n, \quad x, y \in \mathbb{Z}_n.$$

*Vrijednosti $n$ i $e$ su javne, dok su $p$, $q$ i $d$ tajne. Odnosno, uređeni par $(n, e)$ je javni, dok je $(p, q, d)$ tajni ključ.*

U prethodnoj definiciji s $\varphi(n)$ smo označili Eulerovu funkciju. Eulerova funkcija je broj elemenata skupa $\{1, \ldots, n\}$ koji su relativno prosti s $n$. Kako je $n = pq$ pri čemu su $p$ i $q$ prosti brojevi, u našem slučaju vrijedi:

$$\varphi(n) = \varphi(pq) = \varphi(p)\varphi(q) = (p-1)(q-1)$$

Na idućem primjeru pokazat ćemo kako se vrši šifriranje i dešifriranje u RSA kriptosustavu.

**Primjer 1.1.** *Marko želi poslati Ivi poruku $x = 23$. Odluči se uzeti parametre $p = 5$ i $q = 17$. Tada dobijemo $n = 85$ te $\varphi(n) = 64$. Kako $e$ mora biti relativno prosto s $\varphi(n)$ Marko odlučuje da je $e = 5$. Iz uvjeta $de \equiv 1 \mod \varphi(n)$, lako se izračuna da je $d = 13$. Sada znamo da je javni ključ $(n, e) = (85, 5)$, dok je $(p, q, d) = (5, 17, 13)$ tajni ključ. Marko prvo poruku treba šifrirati, a to će napraviti tako da izračuna $e_K(x) = 23^5 \mod 85$:*

$$23^5 = 23^2 \cdot 23^3 \mod 85 = 19 \cdot 12 \mod 85 = 58 \mod 85.$$

*Iva primi poruku $y = e_K(x) = 58$ koju joj je Marko poslao. Kako Iva zna da je $d = 13$ dešifrirat će poruku koju je dobila:*

$$x = d_K(y) = 58^{13} \mod 85 = 58 \cdot 58^2 \cdot 58^4 \cdot 58^6 \mod 85 = 58 \cdot 49 \cdot 21 \cdot 9 = 23 \mod 85.$$

*Iva je uspješno dešifrirala poruku te je dobila $x = 23$.*

Nakon što smo se upoznali s RSA kriptosustavom možemo se pozabaviti RSA shemom potpisa. Vratimo li se na primjer Ive i Marka, poslanu poruku $x$ Marko može potpisati pomoću RSA sheme potpisa, preciznije, koristeći RSA pravilo dešifriranja $d_K$.

Pošto je $d_K = sig_K$ privatno, jedino Marko može kreirati potpis. Algoritam provjere koristi RSA pravilo šifriranja $e_K$. Kako je $e_K$ javan bilo tko može provjeriti je li kreirani potpis validan.Također, bilo tko može krivotvoriti Markov potpis izabirući slučajan $y$ te izračunavajući $x = e_K(y)$. U tom slučaju $y = sig_K(x)$ bi bio validan potpis za poruku $x$. Međutim, RSA kriptosustav bi bio jako nesiguran kad bi neka treća osoba uspjela odgonetnuti poruku $x$ te zatim izračunati odgovarajući potpis $y$.

**Definicija 1.4.** *Neka vrijedi $n = pq$ pri čemu su $p$ i $q$ prosti brojevi, te $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$. Neka je*

$$\mathcal{K} = \{(n, p, q, d, e) : n = pq, \quad de \equiv 1 \pmod{\varphi(n)}\}.$$

*Za $K \in \mathcal{K}$ definiramo:*

$$sig_K(x) = x^d \mod n,$$
$$ver_K(x, y) = istina \iff x = y^e \mod n, \quad x, y \in \mathbb{Z}_n.$$

*Vrijednosti $n$ i $e$ su javne, dok su $p$, $q$ i $d$ tajne. Odnosno, uređeni par $(n, e)$ je javni, dok je $(p, q, d)$ tajni ključ.*

Na idućem primjeru ilustrirat ćemo kako funkcionira RSA shema potpisa.

**Primjer 1.2.** *Nakon što joj je uspješno poslao RSA šifriranu poruku, Marko ovaj put želi poslati Ivi istu poruku samo što će ovaj put poruka biti i potpisana. Iskoristit ćemo iste vrijednosti kao u prošlom primjeru, odnosno $n = 85, p = 5, q = 17, e = 5, d = 13, x = 23$. Potpis dobijemo tako da izračunamo:*

$$y = sig_K(x) = x^d \mod n$$

*Odnosno:*

$$y = 23^{13} \mod 85 \equiv 78 \mod 85.$$

*Iva dobije uređeni par $(x, y) = (23, 78)$ te se želi uvjeriti da je poruka koju je dobila zaista poslao Marko. To može učiniti tako da provjeri vrijedi li zaista kongruencija:*

$$x \equiv y^e \pmod{n}.$$

*Uvrstimo li vrijednosti, dobijemo da zaista vrijedi:*

$$78^5 \mod 85 = 23 \mod 85.$$

*Time je utvrđeno kako je zaista Marko poslao poruku.*

U prošlom je primjeru Marko poslao Ivi potpisanu poruku koja nije bila šifrirana. No, to je vrlo nesigurna opcija jer bilo tko može presresti tu poruku. Upravo zbog toga mnogo je bolja opcija poslati šifriranu potpisanu poruku. U tom bi slučaju Marko prvo pomoću javnog ključa šifrirao poruku i potpis, a onda ih poslao Ivi. Iva bi zatim prvo pomoću tajnog ključa dešifrirala poruku i potpis, a nakon toga bi se pomoću algoritma provjere potpisa mogla uvjeriti da je zaista Marko taj koji je poslao poruku. U idućem primjeru ilustrirat ćemo taj scenarij.

**Primjer 1.3.** *Marko se ovaj put odluči poslati šifriranu potpisanu poruku Ivi s parametrima* $p = 3$ *i* $q = 23$. *Uz tako odabrane* $p$ *i* $q$ *slijedi* $n = 69$, $\varphi(n) = 44$. *Kako prema definiciji parametar* $e$ *treba biti relativno prost s* $\varphi(n)$, *Marko izabire* $e = 9$. *Iz uvjeta* $de \equiv 1 \pmod{\varphi(n)}$ *slijedi* $d = 5$. *Poruka koju će Marko potpisati i šifrirati, a zatim i poslati Ivi je* $x = 12$. *Potpis ćemo dobiti tako što ćemo izračunati* $x^d \mod n$, *tj:*

$$y = 12^5 \mod 69 = 18 \mod 69$$

*Sada kad smo izračunali potpis, još nam preostaje šifrirati poruku i potpis, a to ćemo napraviti tako što ćemo izračunati* $e_K(x)$ *i* $e_K(y)$*:*

$$e_K(x) = 12^9 \mod 69 = 27 \mod 69$$
$$e_K(y) = 18^9 \mod 69 = 12 \mod 69.$$

*Ivi pošaljemo šifrirani par* $(27, 12)$, *a ona će ga pomoću tajnog ključa* $d = 5$ *dešifrirati tako što će izračunati:*

$$x = d_K(x) = 27^5 \mod 69 = 12 \mod 69$$
$$y = d_K(y) = 12^5 \mod 69 = 18 \mod 69.$$

*Nakon što je uspješno dešifrirala poruku koju joj je Marko poslao, Iva još želi provjeriti je li potpis u poruci odgovara Markovom potpisu. Provjerit će vrijedi li kongruencija* $x \equiv y^e \pmod{n}$. *Nakon što uvrstimo vrijednosti, dobijemo:*

$$18^9 \mod 69 = 12 \mod 69.$$

*Iva je potvrdila kako je poruka koju je dobila potpisana Markovim potpisom.*

Neke od shema potpisa imaju sličnosti s RSA shemom, a jedna od njih je i Rabinova shema potpisa s kojom ćemo se pozabaviti u nastavku.

# 2. Rabinova shema potpisa

U sljedećem potpoglavlju analizirat ćemo Rabinov kriptosustav, a zatim i Rabinovu shemu potpisa. Rabinov kriptosustav nastao je 1979. godine, a dobio je ime po svom izumitelju Michaelu O. Rabinu. Za razliku od RSA kriptosustava koji se zasniva na problemu faktorizacije, Rabinov kriptosustav se zasniva na problemu računanja kvadratnog korijena u $\mathbb{Z}_n$. Taj je problem usko povezan s problemom faktorizacije pa zbog toga kažemo da ovaj kriptosustav ima sličnosti s RSA kriptosustavom. Prije nego definiramo Rabinov kriptosustav, prokomentirat ćemo problem računanja kvadratnog korijena u $\mathbb{Z}_n$.

Neka su $p$ i $q$ prosti brojevi te $n = pq$. Za $y$, $1 \leq y < n$, kažemo da je kvadratni ostatak modulo $n$ ukoliko postoji $x \in \mathbb{Z}_n$ takav da vrijedi $x^2 \equiv y \pmod{n}$. Postoje algoritmi koji efikasno rješavaju kongruenciju $x^2 \equiv y \pmod{p}$ (poglavlje 5.1.7 u [1]). Zanimljivo je da je algoritam posebno jednostavan u slučaju ako vrijedi $p \equiv 3 \pmod{4}$. Tada je rješenje $x \equiv \pm y^{(p+1)/4} \pmod{p}$. Kako postoje dva rješenja kongruencije $x^2 \equiv y \pmod{p}$ te dva rješenja kongruencije $x^2 \equiv y \pmod{q}$, prema Kineskom teoremu o ostatcima dobivamo četiri rješenja kongruencije $x^2 \equiv y \pmod{pq}$. Definirajmo sada Rabinov kriptosustav.

**Definicija 1.5.** *Neka vrijedi $n = pq$, pri čemu su $p$ i $q$ prosti brojevi takvi da vrijedi $p \equiv q \equiv 3 \pmod{4}$ te $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$. Neka je*

$$\mathcal{K} = \{(n, p, q) : n = pq\}.$$

*Za $K \in \mathcal{K}$ definiramo:*

$$e_K(x) = x^2 \mod n,$$
$$d_K(y) = \sqrt{y} \mod n, \quad x, y \in \mathbb{Z}_n.$$

*Vrijednost $n$ je javna, dok su $p$ i $q$ tajne. Odnosno, $n$ je javni, dok je uređeni par $(p, q)$ tajni ključ.*

Funkcija $e_K$ nije injekcija što je jedan od nedostataka Rabinovog kriptosustava. Već smo spomenuli da postoje četiri kvadratna korijena modulo $n$ te je upravo zbog toga dešifriranje nemoguće provesti jednoznačno. Postoje razni načini za rješavanje tog problema, a jedan od njih je da se u otvoreni tekst još doda i neka pravilnost, npr. zadnjih nekoliko znakova se ponovi.

**Primjer 1.4.** *Nakon što je primila Markovu poruku, Iva mu želi odgovoriti. Iva ipak preferira Rabinov kriptosustav u odnosu na RSA pa će poruku šifrirati u tom kriptosustavu. Prije nego pošalje šifriranu poruku Marku, Iva mu kaže da je suma znamenaka poruke jednaka* 5. *Kako su* $p$ *i* $q$ *prosti brojevi koji zadovoljavaju* $p \equiv q \equiv 3 \pmod 4$, *Iva odabire* $p = 7$ *te* $q = 19$, *odnosno* $n = 133$. *Poruka koju želi šifrirati je* $x = 23$. *Iz* $e_K(x) = x^2 \mod n$ *slijedi da je:*

$$e_K(x) = 23^2 \mod 133 = 130 \mod 133.$$

*Marko primi poruku* $y = 130$ *te ju želi dešifrirati, odnosno odrediti* $x = d_K(y) = \sqrt{y} \mod n$ *tj.* $x^2 = y \mod n$. *Kako Marko zna tajne* $p$ *i* $q$ *za koje smo već spomenuli da vrijedi* $p \equiv q \equiv 3 \pmod 4$, *tada ujedno i vrijedi* $x \equiv \pm y^{(a+1)/4} \pmod a$ *za* $a \in \{p, q\}$. *Nakon što uvrstimo* $p = 7$ *i* $q = 19$ *dobijemo:*

$$x \equiv 130^2 \pmod 7 \equiv \pm 2 \pmod 7,$$
$$x \equiv 130^5 \pmod{19} \equiv \pm 4 \pmod{19}.$$

*Primjenom Kineskog teorema o ostatcima slijedi da se moguće vrijednosti za* $x$ *nalaze u skupu* $\{23, 61, 72, 110\}$. *Marko zna da je suma znamenaka dešifrirane poruke jednaka* 5 *te otkriva da je dešifrirana poruka* $x = 23$.

Nakon što smo se upoznali s Rabinovim kriptosustavom te vidjeli kako se šifrira, odnosno dešifrira, upoznat ćemo se s Rabinovom shemom potpisa. Konkretno, u prošlom primjeru Iva je koristila Rabinov kriptosustav kako bi šifrirala poruku, a ukoliko želi potpisati tu šifriranu poruku moći će to napraviti koristeći Rabinovu shemu potpisa, odnosno njegovo pravilo dešifriranja $d_K$. Kako smo već rekli, dešifriranje nije jednoznačno određeno. Kao štp postoje četiri moguća rješenja, slično je i kod potpisivanja poruke. Iva će moći izračunati sva četiri moguća potpisa te po svojoj želji odabrati jedan od njih koji će uz poruku poslati Marku.

**Definicija 1.6.** *Neka vrijedi* $n = pq$, *pri čemu su* $p$ *i* $q$ *prosti brojevi takvi da vrijedi* $p \equiv q \equiv 3 \pmod 4$ *te* $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$. *Neka je*

$$\mathcal{K} = \{(n, p, q) : n = pq\}.$$

*Za* $K \in \mathcal{K}$ *definiramo:*

$$sig_K(x) = \sqrt{x} \mod n,$$
$$ver_K(x, y) = istina \iff x = y^2 \mod n, \quad x, y \in \mathbb{Z}_n.$$

*Vrijednost* $n$ *je javna, dok su* $p$ *i* $q$ *tajne. Odnosno,* $n$ *je javni, dok je uređeni par* $(p, q)$ *tajni ključ.*

Iva se odluči poslati Marku šifriranu potpisanu poruku, a na idućem primjeru ćemo pokazati kako to zaista izgleda.

**Primjer 1.5.** *Iva javi Marku da će mu poslati šifriranu potpisanu poruku kojoj je umnožak znamenaka jednak* 12. *Iva ovaj put odabire* $p = 11$ *te* $q = 19$, *odnosno* $n = 209$. *Poruka koju želi šifrirati je* $x = 34$. *Iz* $e_K(x) = x^2 \mod n$, *slijedi da je*

$$e_K(x) = 34^2 \mod 209 = 111 \mod 209$$

. *Da bi potpisala poruku Iva treba izračunati kvadratne korijene te izabrati željeni potpis. Kako vrijedi* $x \equiv \pm y^{(a+1)/4} \pmod a$ *za* $a \in \{p, q\}$, *slijedi:*

$$\begin{aligned} x &\equiv 111^3 \pmod{11} \equiv \pm 1 \pmod{11}, \\ x &\equiv 111^5 \pmod{19} \equiv \pm 4 \pmod{19}. \end{aligned}$$

*Kada primijenimo Kineski teorem o ostatcima, slijedi da se za potpis poruke* $x$ *može odabrati bilo koja vrijednost iz skupa* $\{23, 34, 175, 186\}$. *Iva odluči za potpis odabrati vrijednost* 175, *odnosno Iva pošalje Marku šifriranu potpisanu poruku* $(111, 175)$. *Prije nego dešifrira poruku Marko se želi uvjeriti da je zaista dobio Ivinu poruku, tj. želi provjeriti vrijedi li kongruencija* $x \equiv y^2 \mod n$. *Nakon što uvrstimo, dobijemo:*

$$\begin{aligned} 111 &= 111 \mod 209, \\ 175^2 &= 111 \mod 209. \end{aligned}$$

*Zaista, poruka koju je Iva dobila ima Markov potpis. Dešifrirajmo sada poruku* $y = 111$, *tj. odredimo* $x = d_K(y) = \sqrt{y} \mod n$ *tj.* $x^2 = y \mod n$. *Kako Marko zna tajne* $p$ *i* $q$ *za koje smo već spomenuli da vrijedi* $p \equiv q \equiv 3 \pmod 4$, *tada ujedno i vrijedi* $x \equiv \pm y^{(a+1)/4} \pmod a$ *za* $a \in \{p, q\}$. *Nakon što uvrstimo* $p = 11$ *i* $q = 19$ *dobijemo:*

$$\begin{aligned} x &\equiv 111^3 \pmod{11} \equiv \pm 1 \pmod{11}, \\ x &\equiv 111^5 \pmod{19} \equiv \pm 4 \pmod{19}. \end{aligned}$$

*Nakon što primijenimo Kineski teorem o ostatcima, slijedi da se moguće vrijednosti za* $x$ *nalaze u skupu* $\{23, 34, 175, 186\}$. *Marko zna da je umnožak znamenaka dešifrirane poruke jednak* 12 *te otkriva da je dešifrirana poruka* $x = 34$.

# Poglavlje 2

# DSA

DSA (eng. Digital Signature Algorithm) je algoritam za generiranje sheme potpisa. Originalno je nastao 1991. godine od strane NSA (eng. National Security Agency). Međutim, predloženi potpis je 1991. godine doživio mnoge kritike. Jedna od kritika je bila što NIST-ov (eng. National Institute of Standards and Technology) postupak odabira nije bio javan. Od tehničkih zamjerki izdvaja se činjenica što je prosti broj $p$ bio fiksiran na 512 bita. Mnogi su predlagali da je bolje da veličina od $p$ ne bude fiksna, a na kraju je to i usvojeno. Konačno je u prosincu 1994. godine usvojen te je propisan takozvanim DSS-om (eng. Digital Signature Standard) od strane NIST-a.

Ovaj algoritam je kreiran za potrebe potpisa, za razliku od recimo RSA i Rabinovog kriptosustava koji se koriste za šifriranje te su izmjenjeni za potrebe sheme potpisa. DSA je razvijen po uzoru na ElGamalov kriptosustav, odnosno ElGamalovu shemu potpisa. Da bi razumjeli način na koji funkcionira DSA prvo ćemo se pozabaviti ElGamalovom shemom potpisa.

## 1. ElGamalova shema potpisa

Prije nego krenemo na ElGamalovu shemu potpisa definirat ćemo problem diskretnog logaritma, a zatim ćemo nešto reći i o ElGamalovom kriptosustavu. Razlog je taj jer je upravo na kompliciranosti tog problema zasnovan ElGamalov kriptosustav, odnosno ElGamalova shema potpisa.

**Definicija 2.1.** *Neka je $(G, *)$ konačna grupa te neka je $g \in G$. Neka je $H = \{g^a : a \geq 0\}$ podgrupa od $G$ generirana s $g$ i $h \in H$. Diskretnim logaritmom nazivamo najmanji nenegativni cijeli broj $x$ za koji vrijedi $h = g^x$ pri čemu je:*

$$\underbrace{g * g * \cdots * g}_{x \text{ puta}} = g^x$$

*Diskretni logaritam označavamo s $\log_g h$.*

ElGamalov kriptosustav je zasnovan na problemu računanja diskretnog logaritma u grupi $(\mathbb{Z}_p^*, \cdot_p)$, a nastao je 1985. godine od strane egipatskog kriptografa Tahera ElGamala.

Kako bismo lakše razumjeli ElGamalov kriptosustav, prvo ćemo definirati neke osnovne pojmove poput reda te primitivnog korijena koje ćemo dalje koristiti u radu.

**Definicija 2.2.** *Neka je $n \in \mathbb{N}$ i $x \in \mathbb{Z}$ pri čemu vrijedi $(n, x) = 1$. Red od $x$ modulo $n$ je najmanji broj $a \in \mathbb{N}$ takav da je:*

$$x^a \equiv 1 \pmod{n}.$$

*Red od $x$ modulo $n$ označavamo s $ord_n(x)$.*

**Definicija 2.3.** *Neka je $n \in \mathbb{N}$ i $x \in \mathbb{Z}$ pri čemu vrijedi $(n, x) = 1$. Za $x$ kažemo da je primitivni korijen modulo $n$ ako vrijedi:*

$$ord_n(x) = \varphi(n).$$

Vrijedi napomenuti kako svaki prost broj $p$ ima svoj primitivni korijen modulo $p$. Sada kad smo se upoznali i s tim pojmovima možemo definirati ElGamalov kriptosustav.

**Definicija 2.4.** *Neka je $p$ prost broj i $\alpha \in \mathbb{Z}_p^*$ primitivni korijen modulo $p$. Neka je $\mathcal{P} = \mathbb{Z}_p^*, \mathcal{C} = \mathbb{Z}_p^* \times \mathbb{Z}_p^*$ i neka je:*

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}.$$

*Za $K \in \mathcal{K}$ i tajni slučajni broj $k \in \mathbb{Z}_{p-1}$ definiramo:*

$$e_K(x, k) = (y_1, y_2)$$

*pri čemu su:*

$$y_1 = \alpha^k \mod p,$$
$$y_2 = x\beta^k \mod p.$$

*Za* $y_1, y_2 \in \mathbb{Z}_p^*$ *definiramo:*

$$d_K(y1, y2) = y_2(y_1^a)^{-1} \mod p.$$

*Vrijednosti* $p, \alpha, \beta$ *su javne, dok je* $a$ *tajna. Odnosno, uređena trojka* $(p, \alpha, \beta)$ *je javni, dok je* $a$ *tajni ključ.*

Kako bi $a$ zaista bio tajni ključ, $p$ mora biti dovoljno velik te bi tako problem diskretnog logaritma bio praktički nerješiv u $\mathbb{Z}_p^*$. Zbog toga se preporuča korištenje prostih brojeva od barem 1024 bita.

**Primjer 2.1.** *Nakon što je uspješno dešifrirao poruku koju mu je Iva poslala, Marko joj želi odgovoriti. Ovaj put će šifrirati poruku* $x = 56$ *pomoću ElGamalovog kriptosustava. Odabire* $p = 5743$ *i* $a = 635$*. Iz uvjeta definicije da je* $\alpha$ *primitivni korijen modulo* $p$*, Marko se odluči izabrati* $\alpha = 10$*. Kako vrijedi da je* $\beta = \alpha^a \mod p$*, slijedi da je* $\beta = 10^{635} \mod 5743 = 1531$*. Marko odabire slučajni jednokratni ključ* $k = 487$ *te računa* $e_K(x, k)$*, odnosno:*

$$y_1 = 10^{487} \mod 5743 = 4430,$$
$$y_2 = 56 \cdot 1531^{487} \mod 5743 = 3105.$$

*Nakon što je primila Markovu šifriranu poruku* $(4430, 3105)$*, Iva ju dešifrira, tj. računa* $d_K(y1, y2) = y_2(y_1^a)^{-1} \mod p$*. Uvrštavanjem navedenih vrijednosti u prethodnu jednadžbu slijedi:*

$$x = 3105(4430^{635})^{-1} \mod 5743 = 3105 \cdot 158^{-1} \mod 5743 = 3105 \cdot 4907 \mod 5743 = 56$$

*Iva je uspješno dešifrirala poruku te dobila* $x = 56$*.*

Nakon što smo ilustrirali na koji način funkcionira ElGamalov kriptosustav, pozabavit ćemo se ElGamalovom shemom potpisa. Ova shema je, kao i kriptosustav, nedeterministička, što znači da svaka šifrirana poruka ima više validnih potpisa te algoritam provjere $ver_K$ mora autentificirati svaki od tih potpisa. Slično kao i kod RSA sheme potpisa, potpis može bilo tko provjeriti jer algoritam provjere $ver_K$ koristi samo javne informacije, dok algoritam za generiranje potpisa $sig_K$ koristi tajne informacije. To, naravno, ima smisla jer bi u suprotnom svatko mogao krivotvoriti potpis.

**Definicija 2.5.** *Neka je $p$ prost broj takav da je problem diskretnog logaritma u $\mathbb{Z}_p$ teško rješiv i $\alpha \in \mathbb{Z}_p^*$ primitivni korijen modulo $p$. Neka je $\mathcal{P} = \mathbb{Z}_p^*, \mathcal{C} = \mathbb{Z}_p^* \times \mathbb{Z}_{p-1}$ i neka je:*

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}.$$

*Za $K \in \mathcal{K}$ i tajni slučajni broj $k \in \mathbb{Z}_{p-1}^*$ definiramo:*

$$sig_K(x, k) = (\gamma, \delta)$$

*pri čemu su:*

$$\gamma = \alpha^k \mod p,$$
$$\delta = (x - a\gamma)k^{-1} \pmod{p-1}.$$

*Za $x, \gamma \in \mathbb{Z}_p^*$ i $\delta \in \mathbb{Z}_{p-1}$ definiramo:*

$$ver_K(x, (\gamma, \delta)) = istina \iff \beta^\gamma \gamma^\delta \equiv a^x \pmod{p}.$$

*Vrijednosti $p, \alpha, \beta$ su javne, dok je $a$ tajna. Odnosno, uređena trojka $(p, \alpha, \beta)$ je javni, dok je $a$ tajni ključ.*

Nakon što smo ilustrirali šifriranje i dešifriranje pomoću ElGamalovog kriptosustava, u idućem primjeru pokazat ćemo generiranje i provjeru potpisa pomoću ElGamalove sheme potpisa.

**Primjer 2.2.** *Marko želi potpisati poruku $x = 17$ pomoću ElGamalove sheme potpisa. Izabere $p = 491$, $a = 89$ te kako je $\alpha$ primitivan korijen modulo $p$ odabire $\alpha = 2$. Slijedi:*

$$\beta = \alpha^a \mod p = 2^{89} \mod 491 = 439.$$

*Za slučajan $k$ Marko odabere $k = 79$. Tada je:*

$$\gamma = 2^{79} \mod 491 = 443,$$
$$\delta = (17 - 89 \cdot 443) \cdot 459 \mod 490 = 140.$$

*Nakon što je dobila Markov potpis $(443, 144)$, Iva želi provjeriti da slučajno netko nije falsificirao taj potpis. Odnosno, želi provjeriti vrijedi li kongruencija $\beta^\gamma \gamma^\delta \equiv a^x \pmod{p}$. Zaista:*

$$439^{443} \cdot 443^{140} \equiv 466 \pmod{491}$$

*i*

$$2^{17} \equiv 466 \pmod{491}.$$

*Algoritmom provjere Iva je utvrdila da je potpis koji je dobila validan, odnosno dobila je Markov potpis.*

Na prošlom primjeru smo ilustrirali ElGamalovu shemu potpisa, a u nastavku ćemo se pozabaviti DSA shemom potpisa.

# 2. DSA

Nakon što smo predstavili ElGamalovu shemu potpisa možemo se pozabaviti DSA algoritmom koji je razvijen upravo po uzoru na ElGamalovu shemu. ElGamalova shema potpisa nije sigurnija od rješavanja problema diskretnog logaritma. Upravo je to razlog zbog kojeg smo primorani odabrati veliki parametar $p$, kako bi se otežao taj problem. Kao što je već spomenuto, preporuča se korištenje broja $p$ od barem 1024 bita. Ne postoji nikakav propisan standard ili direktiva u vezi korištenja barem 1024 bita, već je to samo preporuka zbog brige oko sigurnosti problema diskretnog logaritma. U tom slučaju ElGamalova shema potpisa sadrži 2048 bita, a to je prilično dugačak potpis. Kako to nije praktično, došlo je do potrebe za skraćivanjem potpisa, odnosno nastaje DSA algoritam.

Kod DSA sheme pojavljuje se $q$ koji je 160-bitni prosti broj dok je $p$ $L$-bitni prosti broj, pri čemu $L$ zadovoljava sljedeća dva uvjeta:

1. $L \equiv 0 \pmod{64}$
2. $512 \leq L \leq 1024$.

Prije nego se potpiše, poruka će biti hashirana pomoću SHA-1 algoritma za hashiranje. SHA-1 (eng. Secure Hash Algorithm 1) je 160-bitni algoritam za hashiranje koji je nastao 1993. godine te je postao jedan od najraširenijih hash algoritama, a više o algoritmu se može pronaći u [1] i [4]. Vratimo li se na DSA, rezultat hashiranja će biti izmijenjena 160-bitna poruka potpisana 320-bitnim potpisom, pri čemu su sve operacije obavljene unutar $\mathbb{Z}_p$ te $\mathbb{Z}_q$.

U ElGamalovoj shemi potpisa $\delta$ je definiran kao:

$$\delta = (x - a\gamma)k^{-1} \pmod{p-1}.$$

Promijenimo li u prethodnom izrazu vrijednost unutar zagrade $x - a\gamma$ u $x + a\gamma$, očito je da će se uvjet kod algoritma provjere $\beta^{\gamma}\gamma^{\delta} \equiv a^{x} \pmod{p}$ promijeniti u:

$$a^{x}\beta^{\gamma} \equiv \gamma^{\delta} \pmod{p}.$$

Neka je $\alpha \in \mathbb{Z}_p^*$ takav da je $\alpha$ $q$-ti korijen od 1 modulo $p$. Tada $\alpha$ ima red $q$, a kako su u ElGamalovoj shemi $\beta$ i $\gamma$ potencije od $\alpha$, tada i za njih vrijedi da su reda $q$. Upravo zbog toga, sve potencije u kongruenciji $a^{x}\beta^{\gamma} \equiv \gamma^{\delta} \pmod{p}$ mogu biti reducirane modulo $q$ bez ikakvog utjecaja na vrijednost kongruencije. Kako se u DSA poruka $x$ zamijeni 160-bitnom hashiranom porukom, pretpostaviti ćemo da je $x \in \mathbb{Z}_q$. Kako bi vrijedilo da je $\delta \in \mathbb{Z}_q$, izmijenit ćemo $\delta$ tako da vrijedi:

$$\delta = (x + a\gamma)k^{-1} \mod q.$$

Ostaje nam još promotriti izraz $\gamma = \alpha^{k} \mod p$. Definirajmo $\gamma'$ kao:

$$\gamma' = \gamma \mod q = (\alpha^{k} \mod p) \mod q.$$

Primijetimo kako vrijedi:

$$\delta = (x + a\gamma')k^{-1} \mod q.$$

Također, u uvjet kod algoritma provjere možemo uvrstiti $\gamma'$ ali samo kao eksponent od $\beta$. Tada imamo:

$$a^{x}\beta^{\gamma'} \equiv \gamma^{\delta} \pmod{p}.$$

Pretpostavimo da je $\delta \neq 0$ te obje strane prethodne kongruencije potencirajmo s $\delta^{-1}$. Sada imamo:

$$a^{x\delta^{-1}}\beta^{\gamma'\delta^{-1}} \mod p = \gamma.$$

Prethodnu jednakost možemo skratiti modulo $q$ nakon čega dobivamo:

$$\gamma' = (a^{x\delta^{-1}}\beta^{\gamma'\delta^{-1}} \mod p) \mod q.$$

Sada ćemo definirati DSA shemu potpisa pri čemu ćemo iskoristiti prethodno izvedene jednadžbe. Poruku $x$ ćemo zamijeniti s SHA-1($x$), dok ćemo $\gamma'$ ubuduće označavati s $\gamma$.

**Definicija 2.6.** *Neka je $p$ $L$-bitni prosti broj takav da je problem diskretnog logaritma u $\mathbb{Z}_p$ teško rješiv te $L \equiv 0 \pmod{64}$ i $512 \leq L \leq 1024$. Neka je $q$ 160-bitni prosti broj pri čemu je $q$ djelitelj od $p-1$. Neka je $\alpha \in \mathbb{Z}_p^*$ $q$-ti korijen od 1 modulo $p$ te neka je $\mathcal{P} = \{0,1\}^*, \mathcal{C} = \mathbb{Z}_q^* \times \mathbb{Z}_q^*$ i*

$$\mathcal{K} = \{(p, q, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\},$$

*pri čemu vrijedi $0 \leq a \leq q-1$.*
*Za $K \in \mathcal{K}$ i za slučajni odabrani broj $k, 1 \leq k \leq q-1$ definiramo:*

$$sig_K(x,k) = (\gamma, \delta),$$

*pri čemu su:*

$$\gamma = (\alpha^k \mod p) \mod q,$$
$$\delta = (SHA\text{-}1(x) - a\gamma)k^{-1} \mod q.$$

*U slučaju da je $\gamma = 0$ ili $\delta = 0$ $k$ treba biti ponovno odabran. Za $x \in \{0,1\}^*$ i $\gamma, \delta \in \mathbb{Z}_q^*$ potpis se može provjeriti računajući:*

$$e_1 = SHA\text{-}1(x)\delta^{-1} \mod q,$$
$$e_2 = \gamma\delta^{-1} \mod q,$$
$$ver_K(x,(\gamma,\delta)) = istina \iff (\alpha^{e_1}\beta^{e_2} \mod p) \mod q = \gamma.$$

*Vrijednosti $p, q, \alpha, \beta$ su javne, dok je $a$ tajna. Odnosno, uređena četvorka $(p, q, \alpha, \beta)$ je javni, dok je $a$ tajni ključ.*

Na idućem primjeru ilustrirat ćemo DSA potpisivanje tako što će Iva poslati Marku poruku potpisanu pomoću DSA sheme. Iva će za potpisivanje odabrati puno manje parametre $p$ i $q$ nego što se to zapravo preporuča. No, prvo primijetimo da ukoliko Iva izračuna $\delta \equiv 0 \pmod{q}$, trebala bi ga odbaciti te krenuti s algoritmom ispočetka i trebala bi odabrati novi broj $k$. Međutim, malo je vjerojatno da će se u stvarnom slučaju to zaista dogoditi. Razlog tomu je što je vjerojatnost događaja $\delta \equiv 0 \pmod{q}$ reda $2^{-160}$ što je jako mali broj.

**Primjer 2.3.** *Iva želi poslati Marku poruku $x$ čija je SHA-1 vrijednost 27, tj. SHA-1$(x) = 27$. Odabire $q = 59$, a kako prema definiciji mora vrijediti $p - 1 \equiv 0 \pmod{q}$ Iva se odluči za $p = 3541$ jer je $3541 = 60q + 1$. Kako je 7 primitivni element u $\mathbb{Z}_{3541}^*$, a $\alpha$ $q$-ti korijen od 1 modulo $p$, slijedi da je:*

$$\alpha = 7^{60} \mod 3541 = 3499.$$

*Budući da mora vrijediti* $0 \leq a \leq q-1$, *Iva odabire* $a = 34$ *te je tada:*

$$\beta = \alpha^a \mod p = 3499^{34} \mod 3541 = 3088.$$

*Za slučajno odabrani* $k = 41$, *Iva izračuna* $k^{-1} \mod 59 = 36$. *Sada možemo odrediti* $\gamma$ *i* $\delta$*:*

$$\gamma = (3499^{41} \mod 3541) \mod 59 = 2386 \mod 59 = 26,$$
$$\delta = (27 - 34 \cdot 26) \cdot 36 \mod 59 = 5.$$

*Nakon što je Marko dobio poruku s Ivinim potpisom* $(26, 5)$ *želi provjeriti je li potpis zbilja Ivin. Marko će to učiniti tako što će provjeriti je li zadovoljena tražena jednakost:*

$$(\alpha^{e_1}\beta^{e_2} \mod p) \mod q = \gamma,$$

*pri čemu su:*

$$e_1 = \textit{SHA-1}(x)\delta^{-1} \mod q,$$
$$e_2 = \gamma\delta^{-1} \mod q.$$

*No, prije nego uvrsti vrijednosti Marko računa* $\delta^{-1} \mod q = 5^{-1} \mod 59 = 12$. *Sada slijedi:*

$$e_1 = 27 \cdot 12 \mod 59 = 29,$$
$$e_2 = 28 \cdot 12 \mod 59 = 41.$$

*Konačno, Marko želi provjeriti vrijedi li* $(\alpha^{e_1}\beta^{e_2} \mod p) \mod q = 26$. *Uvrštavanjem dobija:*

$$(3499^{29} \cdot 3088^{41} \mod 3541) \mod 59 = 26 \mod 59 = 26$$

*te se tako uvjeri kako je poruku zaista potpisala Iva.*

# Poglavlje 3

# Nepobitni potpisi

Nepobitna shema potpisa uvedena je 1989. godine od strane Davida Chauma i Hansa van Antwerpena. Ova shema ima nekoliko novih značajki, a jedna od glavnih je da primatelj ne može provjeriti je li poruku koju je dobio zaista potpisala osoba koja ju je poslala bez njene suradnje. Jedna od prednosti ove značajke je svakako to što štiti pošiljatelja od mogućnosti da se dokumenti koje je potpisao dupliciraju i šalju elektronički bez njezinog dopuštenja. Provjera se vrši tzv. protokolom izazov-reakcija (engl. challenge-and-response protokol).

Međutim, kako je suradnja pošiljatelja potrebna za provjeru potpisa, pitanje je što pošiljatelja spriječava od odbacivanja potpisa kojeg je napravio ranije. Odnosno, pošiljatelj može tvrditi kako je validan potpis krivotvoren tako što će odbiti sudjelovati u protokolu provjere $ver_K$ ili tako što će davati lažne informacije kako bi na kraju protokol provjere rezultirao zaključkom da se radi o krivotvorenju. Kako bi se to spriječilo nepobitna shema potpisa se sastoji još i od protokola poricanja pomoću kojeg pošiljatelj može dokazati da je potpis krivotvoren. Pomoću ovog algoritma pošiljatelj može stvarno dokazati da potpis nije njegov, odnosno da je krivotvoren. Ukoliko odbije sudjelovati u tom procesu, može se reći da je potpis validan.

Nadalje, nepobitna shema potpisa sastoji se od tri komponente: algoritma potpisivanja $sig_K$, protokola provjere $ver_K$ i protokola poricanja. Prvo ćemo definirati algoritam potpisivanja i protokol provjere Chaum-van Antwerpenove sheme potpisa, dok ćemo nešto kasnije reći malo više o protokolu poricanja.

**Definicija 3.1.** *Neka je $p = 2q + 1$ prost broj takav da je $q$ prost te da je problem diskretnog logaritma teško rješiv u $\mathbb{Z}_p^*$. Neka je $\alpha \in \mathbb{Z}_p^*$ element reda $q$. Neka je $1 \leq a \leq q-1$ i definiramo $\beta = \alpha^a \mod p$. Označimo s $G$ multiplikativnu podgrupu od $\mathbb{Z}_p^*$ reda $q$ ($G$ se sastoji od kvadratnih ostataka modulo $p$). Neka je $\mathcal{P} = \mathcal{C} = G$ i neka*

*je:*

$$\mathcal{K} = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \pmod{p}\}.$$

*Za $K \in \mathcal{K}$ i $x \in G$ definiramo:*

$$y = sig_K(x) = x^a \mod p.$$

*Za $x, y \in G$ provjera se može obaviti izvršavanjem idućeg protokola:*

1. *Primatelj na slučajan način odabire $e_1, e_2 \in \mathbb{Z}_q$.*
2. *Primatelj računa $c = y^{e_1}\beta^{e_2} \mod p$ i šalje izračunatu vrijednost pošiljatelju.*
3. *Pošiljatelj računa $d = c^{a^{-1} \bmod q} \mod p$ i šalje primatelju izračunatu vrijednost.*
4. *Primatelj prihvaća pošiljateljev potpis $y$ kao validan ako i samo ako vrijedi:*

$$d \equiv x^{e_1}\alpha^{e_2} \pmod{p}.$$

*Vrijednosti $p, \alpha, \beta$ su javne, dok je $a$ tajna. Odnosno, uređena trojka $(p, \alpha, \beta)$ je javni, dok je $a$ tajni ključ.*

Objasnimo prvo uloge od $p$ i $q$ u prošloj definiciji. Svi elementi sheme pripadaju $\mathbb{Z}_p$, no operacije koje radimo trebamo obaviti u multiplikativnoj podgrupi $G$ od $\mathbb{Z}_p^*$ prostog reda. Odnosno, moramo biti u mogućnosti izračunati inverze modulo $|G|$ te upravo zbog toga $|G|$ treba biti prost. Zgodno je uzeti $p = 2q + 1$ pri čemu je $q$ prost. Tako je $G$ dovoljno velika što je poželjno iz razloga što su poruke i potpisi elementi u $G$.

Pokažimo sada da primatelj zbilja prihvaća validan potpis. U idućim jednakostima i kongruencijama sve eksponente treba reducirati modulo $q$. Znamo da je:

$$\begin{aligned} d &\equiv c^{a^{-1}} \pmod{p} \\ &\equiv y^{e_1 a^{-1}}\beta^{e_2 a^{-1}} \pmod{p}. \end{aligned}$$

Kako je

$$\beta \equiv \alpha^a \pmod{p},$$

slijedi:

$$\beta^{a^{-1}} \equiv \alpha \pmod{p}.$$

Slično,

$$y \equiv x^a \pmod{p}$$

pa dobijemo:

$$y^{a^{-1}} \equiv x \pmod{p}.$$

Uvrstimo li u kongruenciju za $d$ prethodne dvije kongruencije dobit ćemo:

$$d \equiv x^{e_1}\alpha^{e_2} \pmod{p}.$$

Time smo pokazali da je potpis koji primatelj prihvaća zaista validan. Ilustrirajmo sada to na idućem primjeru.

**Primjer 3.1.** *Iva šalje poruku Marku te se odluči izabrati* $p = 719$. *Znamo da* $p$ *mora zadovoljavati jednakost* $p = 2q+1$ *pri čemu je* $q$ *prost. Iz prethodne jednakosti dobijemo da je* $q = 359$ *te smo time zadovoljili uvjet da* $q$ *mora biti prost. Kako je* 2 *primitivni element,* $2^2 = 4$ *je generator grupe* $G$ *koja se sastoji od kvadratnih ostataka modulo* 719. *Uzet ćemo* $\alpha = 4$ *te* $a = 283$. *Tada je:*

$$\beta = \alpha^a \mod p = 4^{283} \mod 719 = 252.$$

*Poruka koju Iva želi potpisati je* $x = 87$. *Sada računa:*

$$y = sig_K(x) = x^a \mod p = 87^{283} \mod 719 = 63.$$

*Marko je primio Ivinu poruku s potpisom* $y = 63$ *i sada želi provjeriti je li taj potpis validan. Slučajno odabrane vrijednosti su:* $e_1 = 67$, $e_2 = 184$. *Sada računa:*

$$c = y^{e_1}\beta^{e_2} \mod p = 63^{67} \cdot 252^{184} \mod 719 = 675,$$

*a potom šalje Ivi izračunatu vrijednost* $c = 593$. *Iva sada treba izračunati* $d$ *te poslati Marku kako bi on mogao provjeriti potpis. Slijedi da je:*

$$d = c^{a^{-1} \bmod q} \mod p = 675^{283^{-1} \bmod 359} \mod 719 = 675^{222} \mod 719 = 549.$$

*Konačno, Marko provjerava valjanost potpisa tako što računa:*

$$x^{e_1}\alpha^{e_2} \mod p = 87^{67} \cdot 4^{184} \mod 719 = 549.$$

*Zaista, izračunata vrijednost* 549 *jednaka je* $d$ *pa Marko zaključuje da je potpis koji je dobio validan.*

Preostaje nam još pokazati kako pošiljatelj ne može prevariti primatelja da nevažeći potpis prihvati kao validan, a u situaciji kad jeste to moguće vjerojatnost tog događaja je veoma mala. Sljedeći teorem će nam pomoći u tome.

**Teorem 3.1.** *Ako vrijedi* $y \not\equiv x^a \pmod{p}$ *tada će primatelj prihvatiti* $y$ *kao validan potpis za poruku* $x$ *s vjerojatnošću najviše* $1/q$. [1]

Nakon što smo nam je prošli teorem pomogao da dokažemo sigurnost nepobitne sheme potpisa, definirat ćemo protokolom poricanja.

**Definicija 3.2.**

1. *Primatelj na slučajan način odabire* $e_1, e_2 \in \mathbb{Z}_q^*$, *te zatim računa* $c = y^{e_1}\beta^{e_2} \mod p$ *i šalje izračunatu vrijednost pošiljatelju.*
2. *Pošiljatelj računa* $d = c^{a^{-1} \bmod q} \mod p$ *i šalje primatelju izračunatu vrijednost.*
3. *Primatelj provjerava da je* $d \not\equiv x^{e_1}\alpha^{e_2} \pmod{p}$.
4. *Primatelj na slučajan način odabire* $f_1, f_2 \in \mathbb{Z}_q^*$, *te zatim računa* $C = y^{f_1}\beta^{f_2} \mod p$ *i šalje izračunatu vrijednost pošiljatelju.*
5. *Pošiljatelj računa* $D = C^{a^{-1} \bmod q} \mod p$ *i šalje primatelju izračunatu vrijednost.*
6. *Primatelj provjerava da je* $D \not\equiv x^{f_1}\alpha^{f_2} \pmod{p}$.
7. *Primatelj zaključuje da je potpis* $y$ *krivotvoren ako i samo ako je:*

$$(d\alpha^{-e_2})^{f_1} \equiv (D\alpha^{-f_2})^{e_1} \pmod{p}.$$

Kao što vidimo u prošloj definiciji protokol poricanja se sastoji od sedam koraka. Od prvog do trećeg koraka te od četvrtog do šestog koraka izvršavaju se dva neuspješna protokola provjere. Primijetimo, kada bi bilo koji od trećeg, odnosno šestog koraka potvrdio jednakost kongruencije, potpis koji smo dobili bio bi validan. Zadnji korak u protokolu služi za provjeru je li pošiljatelj svoje odgovore ispravno formirao. U ovom trenutku treba dokazati dvije stvari:

1. Pošiljatelj može uvjeriti primatelja da je nevažeći potpis krivotvoren.
2. Pošiljatelj s jako malom vjerojatnošću može uvjeriti primatelja da je validan potpis krivotvoren.

[1] dokaz se može pronaći u [4]

Iduća dva teorema dokazuju ove dvije stvari.

**Teorem 3.2.** *Ako je* $y \not\equiv x^a \pmod{p}$ *te primatelj i pošiljatelj prate protokol poricanja, tada je:*

$$(d\alpha^{-e_2})^{f_1} \equiv (D\alpha^{-f_2})^{e_1} \pmod{p}.$$

**Dokaz.** Prema definiciji znamo:

$$\begin{aligned} d &\equiv c^{a^{-1}} \pmod{p}, \\ c &\equiv y^{e_1}\beta^{e_2} \pmod{p}, \\ \beta &\equiv \alpha^a \pmod{p}. \end{aligned}$$

Sada imamo:

$$\begin{aligned} (d\alpha^{-e_2})^{f_1} &\equiv ((y^{e_1}\beta^{e_2})^{a^{-1}}\alpha^{-e_2})^{f_1} \pmod{p} \\ &\equiv y^{e_1 a^{-1} f_1}\beta^{e_2 a^{-1} f_1}\alpha^{-e_2 f_1} \pmod{p} \\ &\equiv y^{e_1 a^{-1} f_1}\alpha^{e_2 f_1}\alpha^{-e_2 f_1} \pmod{p} \\ &\equiv y^{e_1 a^{-1} f_1} \pmod{p}. \end{aligned}$$

Sličnim računom, koristeći

$$\begin{aligned} D &\equiv C^{a^{-1}} \pmod{p}, \\ C &\equiv y^{f_1}\beta^{f_2} \pmod{p}, \\ \beta &\equiv \alpha^a \pmod{p} \end{aligned}$$

dobijemo da je:

$$(D\alpha^{-f_2})^{e_1} \equiv y^{e_1 a^{-1} f_1} \pmod{p}.$$

Odnosno, slijedi:

$$(d\alpha^{-e_2})^{f_1} \equiv (D\alpha^{-f_2})^{e_1} \pmod{p}$$

pa je zadnji korak kod protokola poricanja zadovoljen. ■

Prošlim teoremom pokazali smo da pošiljatelj ne može uvjeriti primatelja da je nevažeći potpis krivotvoren. Ostaje nam još pokazati da pošiljatelj ne može uvjeriti primatelja da je validan potpis krivotvoren osim s jako malo vjerojatnošću, a za to će nam pomoći sljedeći teorem.

**Teorem 3.3.** *Pretpostavimo da vrijedi* $y \equiv x^a \pmod{p}$ *i da primatelj prati protokol poricanja. Ako vrijedi* $d \not\equiv x^{e_1}\alpha^{e_2} \pmod{p}$ *i* $D \not\equiv x^{f_1}\alpha^{f_2} \pmod{p}$ *tada je vjerojatnost da vrijedi* $(d\alpha^{-e_2})^{f_1} \not\equiv (D\alpha^{-f_2})^{e_1} \pmod{p}$ *jednaka* $1 - 1/q$. [2]

[2] dokaz se može pronaći u [4]

Nakon što smo pokrili sve slučajeve ostaje nam još za pokazati na primjeru kako zaista funkcionira protokol poricanja.

**Primjer 3.2.** *Iva odabire iste parametre kao u prošlom primjeru, odnosno* $p = 719$, $\alpha = 4$, $a = 283$ *te* $\beta = 252$. *Poruku* $x = 87$ *je ovaj put potpisala s izmišljenim potpisom* $y = 124$ *i želi uvjeriti Marka da to nije validan potpis.*

*Marko na slučajan način odabire parametre* $e_1 = 97$, $e_2 = 231$, *te računa:*

$$c = y^{e_1}\beta^{e_2} \mod p = 124^{97} \cdot 252^{231} \mod 719 = 29.$$

*Zatim šalje Ivi vrijednost* $c = 29$ *te ona računa:*

$$d = c^{a^{-1} \bmod q} \mod p = 29^{283^{-1} \bmod 359} \mod 719 = 29^{222} \mod 719 = 464.$$

*Nakon što mu je Iva poslala* $d = 464$, *Marko dobiva:*

$$x^{e_1}\alpha^{e_2} \mod p = 87^{97} \cdot 4^{231} \mod 719 = 496.$$

*Kako je* $464 \neq 496$, *Marko nastavlja s četvrtim korakom protokola poricanja. Izabire parametre* $f_1 = 19$, $f_2 = 158$ *za koje je:*

$$C = y^{f_1}\beta^{f_2} \mod p = 124^{19} \cdot 252^{158} \mod 719 = 697.$$

*Izračunati parametar* $C = 697$ *šalje Ivi koja potom dobiva:*

$$D = C^{a^{-1} \bmod q} \mod p = 697^{283^{-1} \bmod 359} \mod 719 = 697^{222} \mod 719 = 13.$$

*Marko sada računa:*

$$x^{f_1}\alpha^{f_2} \mod p = 87^{19} \cdot 4^{158} \mod 719 = 100.$$

*Kako je* $13 \neq 100$ *Marko dolazi do zadnjeg koraka protokola u kojem provjerava je li potpis* $y = 124$ *krivotvoren, odnosno provjerava:*

$$(d\alpha^{-e_2})^{f_1} \mod p = (464 \cdot 4^{-231})^{19} \mod 719 = 151$$

*i*

$$(D\alpha^{-f_2})^{e_1} \mod p = (13 \cdot 4^{-158})^{97} \mod 719 = 151.$$

*Kako je* $151 = 151$ *Marko zaključuje da je potpis* $y = 124$ *krivotvoren.*

# Poglavlje 4

# Fail-stop potpisi

U ovom poglavlju reći ćemo nešto o fail-stop potpisima. Fail-stop potpisi omogućavaju nam sigurnost protiv jakih krivotvoritelja. U slučaju da krivotvoritelj uspije krivotvoriti pošiljateljev potpis, pošiljatelj će moći, s velikom vjerojatnoću, dokazati da krivotvoriteljev potpis nije validan. Fail-stop shema potpisa konstruirana je 1992. godine od strane Eugene van Heysta i Torbena Prydsa Pedersena. Ova shema potpisa spada pod jednokratne potpise (eng. one-time signatures), odnosno može sigurno potpisati najviše jednu poruku. Sastoji se od algoritma potpisivanja $sig_K$, algoritma provjere $ver_K$ te od fail-stop koncepta. Prvo ćemo definirati algoritam potpisivanja i algoritam provjere van Heyst-Pedersenove sheme, dok ćemo o fail-stop konceptu reći nešto više kasnije.

**Definicija 4.1.** *Neka je $p = 2q + 1$ prost broj takav da je $q$ prost te da je problem diskretnog logaritma teško rješiv u $Z_p^*$. Neka je $\alpha \in \mathbb{Z}_p^*$ element reda $q$. Neka je $1 \leq a_0 \leq q - 1$ i definiramo $\beta = \alpha^{a_0} \mod p$. Vrijednosti $p$, $q$, $\alpha$, $\beta$ i $a_0$ su izabrene od strane nezavisnog (povjerljivog) izvora. Vrijednosti $p$, $q$, $\alpha$ i $\beta$ su javne i smatraju se kao fiksne dok je $a_0$ tajna vrijednost koju ne zna nitko pa čak ni pošiljatelj. Neka je $\mathcal{P} = \mathbb{Z}_q$ i neka je $\mathcal{C} = \mathbb{Z}_q \times \mathbb{Z}_q$. Ključ je oblika:*

$$K = (\gamma_1, \gamma_2, a_1, a_2, b_1, b_2),$$

*pri čemu su $a_1, a_2, b_1, b_2 \in \mathbb{Z}_q$ te:*

$$\gamma_1 = \alpha^{a_1}\beta^{a_2} \mod p$$

*i*

$$\gamma_2 = \alpha^{b_1}\beta^{b_2} \mod p.$$

*Za $K \in \mathcal{K}$ i $x \in \mathbb{Z}_q$ definiramo:*

$$sig_K(x) = (y_1, y_2),$$

*pri čemu su:*

$$y_1 = a_1 + xb_1 \mod q$$

*i*

$$y_2 = a_2 + xb_2 \mod q.$$

*Za $y = (y_1, y_2) \in \mathbb{Z}_q \times \mathbb{Z}_q$ vrijedi:*

$$ver_K(x, y) = istina \iff \gamma_1 \gamma_2^x \equiv \alpha^{y_1} \beta^{y_2} \pmod{p}.$$

*Vrijednosti $\gamma_1, \gamma_2$ su javne, dok su $a_1, a_2, b_1, b_2$ tajne vrijednosti. Odnosno, uređeni par $(\gamma_1, \gamma_2)$ je javni, dok je uređena četvorka $(a_1, a_2, b_1, b_2)$ tajni ključ.*

Iz definicije je jasno da potpis koji je kreirao pošiljatelj zadovoljava uvjet u algoritmu provjere. Pogledajmo sada kako fail-stop svojstvo funkcionira, no prije toga pogledajmo neke potrebne rezultate.

**Definicija 4.2.** *Za ključeve $(\gamma_1, \gamma_2, a_1, a_2, b_1, b_2)$ i $(\gamma_1', \gamma_2', a_1', a_2', b_1', b_2')$ kažemo da su ekvivalentni ukoliko vrijedi $\gamma_1 = \gamma_1'$ i $\gamma_2 = \gamma_2'$.*

**Lema 4.1.** *Neka je $K$ ključ i $y = sig_K(x)$. Tada postoji točno $q$ ključeva $K'$ ekvivalentnih ključu $K$ takvih da je $y = sig_{K'}(x)$.*

**Lema 4.2.** *Neka je $K$ ključ, $y = sig_K(x)$ i $ver_K(x', y') = istina$, pri čemu je $x' \neq x$. Tada postoji jedinstven ključ $K'$ ekvivalentan ključu $K$ takav da je $y = sig_{K'}(x)$ i $y' = sig_{K'}(x')$.*

Prethodne dvije leme kažu da ako je $y$ validan potpis za poruku $x$, postoji $q$ mogućih ključeva koji će potpisati poruku $x$ s $y$. No, za svaku poruku $x' \neq x$ tih $q$ ključeva će dati $q$ različitih potpisa za $x'$. Kao rezultat toga slijedi idući teorem.

**Teorem 4.1.** *Neka je $sig_K(x) = y$ i $x' \neq x$. Tada krivotvoritelj može izračunati $sig_K(x')$ s vjerojatnošću $1/q$.*

Primijetimo da je prethodna vjerojatnost dobijena jer krivotvoritelj ne može znati koji od $q$ mogućih potpisa je koristio pošiljatelj.

Prokomentirajmo sada fail-stop koncept. Do sad smo pokazali da iz danog potpisa $y$ poruke $x$ krivotvoritelj ne može odrediti potpis $y'$ za poruku $x'$. No, razumljivo je da krivotvoritelj može izračunati krivotvoreni potpis $y'' \neq sig_K(x')$ koji može biti verificiran. Međutim ukoliko pošiljatelj dođe do tog krivotvorenog potpisa, moći će sa vjerojatnošću $1 - 1/q$ dokazati da je potpis krivotvoren. Dokaz krivotvorenja je vrijednost $a_0 = \log_\alpha \beta$ koji je poznat samo nezavisnoj osobi iz definicije.

Pretpostavimo da je pošiljatelj upoznat s uređenim parom $(x', y'')$ pri čemu je $ver_K(x', y'') =$ istina i $y'' \neq sig_K(x')$. Tada je:

$$\gamma_1 \gamma_2^{x'} \equiv \alpha^{y_1''} \beta^{y_2''} \pmod{p},$$

gdje je $y'' = (y_1'', y_2'')$. Sada pošiljatelj može izračunati svoj potpis $y' = (y_1', y_2')$ za poruku $x'$. U tom slučaju vrijedi:

$$\gamma_1 \gamma_2^{x'} \equiv \alpha^{y_1'} \beta^{y_2'} \pmod{p}.$$

Sada znamo:

$$\alpha^{y_1''} \beta^{y_2''} \equiv \alpha^{y_1'} \beta^{y_2'} \pmod{p}.$$

Uvrstimo li $\beta = \alpha^{a_0} \mod p$ u prethodnu kongruenciju, dobit ćemo:

$$\alpha^{y_1'' + a_0 y_2''} \equiv \alpha^{y_1' + a_0 y_2'} \pmod{p}$$

ili

$$y_1'' + a_0 y_2'' \equiv y_1' + a_0 y_2' \pmod{q}.$$

Pojednostavimo li zadnji izraz imat ćemo:

$$y_1'' - y_1' \equiv a_0 (y_2' - y_2'') \pmod{q}.$$

Kako je $y_2' \not\equiv y_2'' \pmod{q}$ zbog toga što je $y'$ krivotvorina. Stoga, $(y_2' - y_2'')^{-1} \mod q$ postoji i vrijedi:

$$a_0 = log_\alpha \beta = (y_1'' - y_1')(y_2' - y_2'')^{-1} \mod q.$$

Naravno, prihvaćanjem dokaza krivotvorenja, pretpostavljamo da pošiljatelj ne zna izračunati diskretni logaritam $log_\alpha \beta$. Već smo rekli da je ova shema potpisa jednokratna, odnosno pošiljateljev ključ $K$ se može lako izračunati ako su dvije različite poruke potpisane njime. Za kraj ćemo ilustrirati na primjeru kako funkcionira fail-stop koncept.

**Primjer 4.1.** *Pretpostavimo da je* $p = 563 = 2 \cdot 281 + 1$. *Element* $\alpha = 2^2 = 4$ *je reda* 281 *u* $\mathbb{Z}^*_{563}$. *Pretpostavimo* $a_0 = 134$, *tada je:*

$$\beta = \alpha^{a_0} \mod p = 4^{134} \mod 563 = 158.$$

*Podsjetimo da je Iva upoznata s vrijednostima* $\alpha$ *i* $\beta$, *ali nije s* $a_0$. *Iva sada računa svoj ključ koristeći* $a_1 = 43$, $a_2 = 218$, $b_1 = 96$ *i* $b_2 = 108$, *tada je:*

$$\gamma_1 = \alpha^{a_1}\beta^{a_2} \mod p = 4^{43} \cdot 158^{218} \mod 563 = 391,$$

*dok je:*

$$\gamma_2 = \alpha^{b_1}\beta^{b_2} \mod p = 4^{96} \cdot 158^{108} \mod 563 = 156.$$

*Iva je upoznata s krivotvorenim potpisom* $(235, 78)$ *za poruku* $x = 141$. *Provjerimo je li krivotvoreni potpis* $(235, 78)$ *validan, odnosno je li zadovoljen uvjet* $\gamma_1\gamma_2^x \equiv \alpha^{y_1}\beta^{y_2} \pmod p$ *kod algoritma provjere. Nakon što uvrstimo, dobijemo:*

$$\gamma_1\gamma_2^x \mod p = 391 \cdot 156^{141} \mod 563 = 349$$

*i*

$$\alpha^{y_1}\beta^{y_2} \pmod p = 4^{235} \cdot 158^{78} \mod 563 = 349.$$

*Zaista, krivotvoreni potpis* $(235, 78)$ *je validan. Provjerimo sada kako izgleda Ivin potpis:*

$$y_1 = a_1 + xb_1 \mod q = 43 + 141 \cdot 96 \mod 281 = 91$$
$$y_2 = a_2 + xb_2 \mod q = 218 + 141 \cdot 108 \mod 281 = 272.$$

*Vidimo da se krivotvoreni potpis razlikuje od Ivinog potpisa* $(91, 272)$. *Izračunajmo sada tajni diskretni logaritam:*

$$a_0 = (235 - 91)(272 - 78)^{-1} \mod 281 = 144 \cdot 239 \mod 281 = 134.$$

*Ovime smo dokazali da je potpis* $(235, 78)$ *krivotvoren.*

# Sažetak

U ovom radu smo se upoznali s digitalnim potpisom, objasnili zašto nam je važan te koje su njegove prednosti. Neki od kriptosustava zasnivaju se na problemu faktorizacije, a najpoznatiji od njih su svakako RSA te Rabinov kriptosustav. Oni su predstavljeni u prvom poglavlju rada. Nakon što smo se upoznali s njima, promotrili smo i istoimene sheme potpisa koje su nastale od tih kriptosustava. U drugom je poglavlju naglasak bio na DSA shemu potpisa koja je nastala prema ElGamalovom kriptosustavu, odnosno ElGamalovoj shemi potpisa. Za razliku od RSA i Rabinovog kriptosustava, ElGamalov kriptosustav je zasnovan na problemu diskretnog logaritma kojeg smo također spomenuli u radu. U sljedeća dva poglavlja definirana je nepobitna shema potpisa te fail-stop shema potpisa. Nepobitna shema potpisa za potrebu provjere traži suradnju potpisnika, dok fail-stop potpisi pružaju dodatnu sigurnost od krivotvorenja. Za sve navedene sheme priložen je ilustrativni primjer na kojem su uočljive karakteristike promotrenog potpisa.

## Ključne riječi

Digitalni potpis, kriptosustav, RSA kriptosustav, Rabinov kriptosustav, ElGamalov kriptosustav, RSA shema potpisa, Rabinova shema potpisa, ElGamalova shema potpisa, DSA, Nepobitna shema potpisa, Fail-stop shema potpisa

# Digital signature methods

# Abstract

In this paper we introduced digital signature, explain its importance and its advantages. Some of the cryptosystem are based on the factorization problem and few of them are RSA and Rabin cryptosystem. They are represented in the first chapter as well as corresponding signature schemes. The basis of the second chapter is the DSA signature scheme which is related to the ElGamal cryptosytem and signature scheme. ElGamal cryptosystem is based on the discrete logarithm problem which is also one of the topics in the paper. In addition is defined an undeniable signature scheme and a fail-stop signature scheme. For every scheme that is mentioned in this paper there is corresponding illustrative example.

## Key words

Digital signature, cryptosystem, RSA cryptosystem, Rabin cryptosystem, ElGamal cryptosystem, RSA signature scheme, Rabin signature scheme, ElGamal signature scheme, DSA, Undeniable signature scheme, Fail-stop signature scheme

# Literatura


[1] A. Dujella, M. Maretić, *Kriptografija*, Element, Zagreb, 2007.

[2] I. Matić, *Uvod u teoriju brojeva*, Odjel za matematiku, Sveučilište J.J. Strossmayera u Osijeku, 2015.

[3] A. Menezes, P. Van Oorchot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[4] D. Stinson, *Cryptography: Theory and Practice, 3rd Edition*, Chapman & Hall/CRC, University of Waterloo, Ontario, 2006.,

# Životopis

Rođen sam 14.01.1994. u Vukovaru. Završio sam Osnovnu školu Tenja u Tenji, a zatim i III. gimnaziju u Osijeku. Nakon završene gimnazije, 2012. godine, upisujem Sveučilišni preddiplomski studij matematike na Odjelu za matematiku u Osijeku. Studij završavan 2016. godine s temom završnog rada Quasi-Newtonove metode pod mentorstvom prof. dr. sc. Kristiana Sabe. Zatim, na Odjelu, upisujem Sveučilišni diplomski studij, smjer Matematika i računarstvo. U listopadu 2012. i 2014. godine sudjelujem IEEEXtreme natjecanju iz programiranja. U tvrtki Mono d.o.o obavljam stručnu praksu tijekom kolovoza 2017. godine.